

**REMARKS**

By this Amendment, Applicants amend claims 1, 4, 6, 11, 12, 17, 19, 23, 26, and 28. Claims 1-9, 11, 12, and 15-31 are pending in this application.

In the Office Action,<sup>1</sup> the Examiner provisionally rejected claims 1, 3, 4, 11, 12, 16, 17, 23, 25, and 26 on the ground of nonstatutory obviousness-type double patenting; objected to claims 1, 11, 12, and 23; and rejected claims 1-9, 11, 12, and 15-31 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,566,319 to Lenz ("*Lenz*") in view of U.S. Patent No. 6,343,296 to Lakhamraju et al. ("*Lakhamraju*").

**I. Provisional Non-Statutory Obviousness-Type Double Patenting Rejection**

The Examiner provisionally rejected claims 1, 3, 4, 11, 12, 16, 17, 23, 25, and 26 over claims 1-6, 8-13, 15-20, and 22-25 of copending U.S. Application No. 10/656,208 on the ground of nonstatutory obviousness-type double patenting. Applicants respectfully traverse the provisional rejection and request that the provisional rejection be held in abeyance.

To the knowledge of the undersigned, U.S. Application No. 10/656,208 is currently pending, and, thus, no double patenting circumstances can arise until a patent is granted. Because a patent has not yet issued from U.S. Application No. 10/656,208,

---

<sup>1</sup> The Office Action contains a number of statements reflecting characterizations of the related art, case law, and the claims. Regardless of whether any such statement is identified herein, Applicants decline to automatically subscribe to any statement or characterization in the Office Action.

Applicants respectfully request that the provisional rejection be held in abeyance and any resolution in the form of a terminal disclaimer or otherwise be deferred.

Applicants further note that M.P.E.P. § 804(I)(B) addresses the situation of two copending applications. The section states that “[t]he ‘provisional’ double patenting rejection should continue to be made by the examiner in each application . . . unless that ‘provisional’ double patenting rejection is the only rejection remaining in at least one of the applications. . . . If a ‘provisional’ nonstatutory obviousness-type double patenting (ODP) rejection is the only rejection remaining in the earlier filed of the two pending applications, . . . the examiner should withdraw that rejection and permit the earlier-filed application to issue as a patent without a terminal disclaimer.” Therefore, Applicants respectfully request that the provisional double patenting rejection be withdrawn if it is the only remaining rejection in this application or U.S. Application No. 10/656,208, and neither application has resulted in a granted patent.

## **II. Objection to Claims 1, 11, 12, and 23**

The Examiner objected to claims 1, 11, 12, and 23, alleging that “the phrase ‘read and/or write’ is improper.” Office Action at 5. Applicants respectfully traverse the objection. However, to advance prosecution, Applicants amend claims 1, 11, 12, and 23 to address the Examiner’s concerns. Accordingly, Applicants respectfully request that the Examiner withdraw the objection to claims 1, 11, 12, and 23.

**III. Rejection of Claims 1-9, 11, 12, and 15-31 under 35 U.S.C. § 103(a)**

Applicants respectfully traverse the rejection of claims 1-9, 11, 12, and 15-31 under 35 U.S.C. § 103(a) as being unpatentable over *Lenz* in view of *Lakhamraju*. A *prima facie* case of obviousness has not been established.

“The key to supporting any rejection under 35 U.S.C. 103 is the clear articulation of the reason(s) why the claimed invention would have been obvious.” M.P.E.P. § 2141(III). “[T]he framework for objective analysis for determining obviousness under 35 U.S.C. 103 is stated in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966). . . . The factual inquiries . . . are as follows:

(A) [Determining the scope and content of the prior art;]

(B) Ascertaining the differences between the claimed invention and the prior art;  
and

(C) Resolving the level of ordinary skill in the pertinent art.”

M.P.E.P. § 2141(II). “Office personnel must explain why the difference(s) between the prior art and the claimed invention would have been obvious to one of ordinary skill in the art.” M.P.E.P. § 2141(III).

Independent claim 1, as amended, recites a method for accessing a data object, comprising, for example, “determining whether another process is attempting to perform a transaction with the data object by determining whether an identifier (ID) of the data object is stored in a second lock object,” “upon determining that the ID is not stored in the second lock object and that another process is not attempting to perform a transaction with the data object, storing the ID in the second lock object,” “determining

whether another process is archiving the data object to a second storage location by determining whether the ID is stored in a first lock object and whether a link to a copy of the data object stored in the second storage location is assigned to the ID, wherein another process stores the ID in the first lock object and assigns the link to the ID while archiving the data object,” “upon determining that the ID is not stored in the first lock object, that the link is not assigned the ID, and that another process is not archiving the data object, performing at least one of a read operation and a write operation on the data object,” and “the data object is locked if the ID is stored in at least one of the first lock object and the second lock object.”

*Lenz* discloses “controlling access to data . . . which is shared by a plurality of processors.” *Lenz*, abstract. *Lenz* further discloses that “[t]he lock file comprises . . . control fields containing access administration information authorizing the processors to access the data when not currently being accessed by another processor.” *Id.*, col. 2, ll. 3-8 (emphasis added). Therefore, *Lenz* discloses determining whether another processor is accessing the data. However, *Lenz* is completely silent with respect to archiving or moving the data from one storage location to another storage location. Accordingly, *Lenz* fails to teach or suggest “determining whether another process is archiving the data object to a second storage location” and “upon determining . . . that another process is not archiving the data object,” as recited in claim 1 (emphases added).

*Lenz* further discloses that “[t]he lock file . . . has . . . control fields for the shared data . . . [and] each control field contains information . . . associated with . . . the

respective shared data." *Lenz*, col. 1, ll. 24-31. That is, for every data that is shared among multiple processors, the lock file contains a control field for each data. The example illustrated in Fig. 3 of *Lenz* includes three data records 007, 008, and 009 that are shared among multiple processors, and lock file 3-1 that contains an SKC and a control field for each of the three data records 007, 008, and 009. However, *Lenz* fails to disclose that the IDs (i.e., 007, 008, and 009) are not stored in lock file 3-1. Therefore, in *Lenz*, the IDs 007, 008, and 009 (associated with the SKCs and control fields and corresponding to the data records) are always stored in the lock file.

Moreover, *Lenz* discloses "a status identification code [(SKC)] for the data to indicate whether another of the processors is currently accessing the data." *Id.*, col. 2, ll. 7-9. For example, "SKC could be 0 . . . , i.e. an access right may be granted immediately" or "SKC could be 1 . . . , i.e. an access right may only be granted after detailed examination." *Id.*, col. 5, ll. 49-53. The value of SKC could also be 2 or 3. See *Lenz*, col. 2, ll. 24-40 and Fig. 6. For example, when "a read request is issued by processor 1 to record 007[,] . . . it is examined . . . whether the status identification code [(SKC)] associated with the control field of record 007 in the lock file is 0." *Id.*, col. 5, line 66 to col. 6, line 3. Therefore, *Lenz* discloses determining whether the data record 007 is locked or not locked by checking the value [0, 1, 2, or 3] of SKC associated with ID 007, not by checking whether the ID [007] of the data record is stored or not stored in the file lock, because, as discussed above, the ID 007 is always stored in the lock file. In the Office Action, the Examiner cites *Lenz*, col. 4, line 67, which states "the status identification code [(SKC)] is updated," as allegedly disclosing "deleting the ID from the

first lock object,” as previously recited in claim 1. Office Action at 8. This is incorrect. Although the value of SKC may be updated (e.g., from 0 to 1), the ID associated with the SKC for record 007 is still stored in the lock file. This ID (e.g., 007, 008, or 009) in the lock file of *Lenz* is neither deleted nor updated. Accordingly, *Lenz* fails to teach or suggest “determining whether an identifier (ID) of the data object is stored in a second lock object,” “upon determining that the ID is not stored in the second lock object,” “determining whether the ID is stored in a first lock object,” and “upon determining that the ID is not stored in the first lock object,” as recited in claim 1 (emphases added).

Moreover, because the lock file in *Lenz* always stores the IDs of data records and because *Lenz* is completely silent with respect to archiving, *Lenz* fails to teach or suggest “determining whether another process is archiving the data object . . . , wherein another process stores the ID in the first lock object . . . while archiving the data object,” as recited in claim 1.

Furthermore, claim 1 recites “a first lock object” and a separate and distinct “second lock object.” However, *Lenz* discloses only one lock file comprising SKCs and control fields for multiple data records 007, 008, and 009. See *Lenz*, Fig. 3. Therefore, *Lenz* also fails to teach or suggest “determining whether another process is attempting to perform a transaction with the data object by determining whether an identifier (ID) of the data object is stored in a second lock object” and “determining whether another process is archiving the data object to a second storage location by determining whether the ID is stored in a first lock object,” as recited in claim 1 (emphases added). Also, because *Lenz* discloses only one lock file, *Lenz* fails to teach or suggest “the data

object is locked if the ID is stored in at least one of the first lock object and the second lock object," as recited in claim 1.

*Lakhamraju* discloses that "for each object, a set of exact parents is found and locked, their references to the migrating object are updated and the migrating object is moved." *Lakhamraju*, col. 4, ll. 23-26. However, *Lakhamraju* does not disclose the specific mechanism used to lock and unlock the parent references. That is, *Lakhamraju* does not disclose that the parent references are locked by "stor[ing] the ID [of the data object] in the first lock object," as recited in claim 1. Therefore, *Lakhamraju* fails to teach or suggest "determining whether another process is archiving the data object to a second storage location by determining whether the ID is stored in a first lock object . . . , wherein another process stores the ID in the first lock object . . . while archiving the data object" and "the data object is locked if the ID is stored in at least one of the first lock object and the second lock object," as recited in claim 1.

*Lakhamraju* further discloses that "an External Reference Table ('ERT') . . . list[s] the objects with at least one parent from another partition." *Lakhamraju*, col. 4, ll. 35-37. However, *Lakhamraju* does not disclose determining whether another process is attempting to perform a transaction with the object by determining whether an ID of the object is listed in the ERT table; upon determining that the ID is not listed in the ERT table, listing the ID in the ERT table; determining whether another process is archiving the object by determining whether the ID is listed in the ERT table; and upon determining that the ID is not listed in the ERT table, performing an operation on the

object. Therefore, the ERT table of *Lakhamraju* cannot correspond to the claimed “first lock object” or the claimed “second lock object.”

Furthermore, *Lakhamraju* discloses that “a Temporary Reference Table (‘TRT’) . . . list[s] separately for each migrating object, a new reference or a deleted reference to that migrating object.” *Lakhamraju*, col. 4, ll. 48-51. However, as shown by pseudo codes in Figs. 1-4 of *Lakhamraju*, objects (migrating objects and new or deleted reference objects) that are listed in the TRT table are both locked and unlocked while the objects are stored in the TRT table. On the contrary, claim 1 recites “the data object is locked if the ID is stored in at least one of the first lock object and the second lock object.” Therefore, the TRT table of *Lakhamraju* cannot correspond to the claimed “first lock object” or the claimed “second lock object.”

At least because *Lakhamraju* does not teach or suggest the claimed “first lock object” and “second lock object,” *Lakhamraju* fails to teach or suggest “determining whether an identifier (ID) of the data object is stored in a second lock object,” “upon determining that the ID is not stored in the second lock object,” “determining whether the ID is stored in a first lock object . . . , wherein another process stores the ID in the first lock object . . . while archiving the data object,” and “upon determining that the ID is not stored in the first lock object,” as recited in claim 1. For at least these reasons, *Lakhamraju* fails to cure all the deficiencies of *Lenz*.

For at least the foregoing reasons, the scope and content of the prior art have not been properly determined, and the differences between the prior art and claim 1 have not been properly ascertained. Moreover, the Examiner has not identified any factors



that would motivate one of ordinary skill in the art to modify the teachings of the prior art to achieve the claimed combination. Accordingly, no reason has been clearly articulated as to why the prior art would have rendered claim 1 obvious to one of ordinary skill in the art. Therefore, a *prima facie* case of obviousness has not been established with respect to claim 1.

Independent claims 11, 12, and 23, although different in scope from claim 1, are allowable for at least reasons similar to those presented above with respect to claim 1. Dependent claims 2-9, 15-22, and 24-31 are allowable at least due to their dependence from an allowable independent claim. Accordingly, Applicants respectfully request that the Examiner withdraw the rejection of claims 1-9, 11, 12, and 15-31 under 35 U.S.C. § 103(a).

### **CONCLUSION**

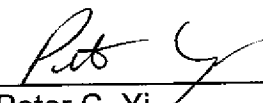
In view of the foregoing, Applicants respectfully request reconsideration of this application and timely allowance of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to Deposit Account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,  
GARRETT & DUNNER, L.L.P.

Dated: June 4, 2009

By:   
Peter C. Yi  
Reg. No. 61,790  
202.408.4485